

Illustrative multilevel focus+context visualization along snaking paths

Jeffrey F. Packer · Mahmudul Hasan · Faramarz F. Samavati

Received: date / Accepted: date

Abstract Artistic anatomical illustrations often focus on cross sections of long, layered, cylindrical structures. Such illustrations emphasize structures along transitions between focal points over a *snaking* path that optimally traverses the span of a limited space. The transitions between focal points form a multilevel visualization hierarchy. In this article, we present an approach to automatically create focus+context visualizations of the described form. First, a method to automatically create a snaking path through space by applying a pathfinding algorithm is presented. A 3D curve is created based on a 2D snaking path. Then we describe a process to deform cylindrical structures in segmented volumetric models along the 3D curve and provide preliminary geometric models as templates for artists to build upon. Our constrained volume sculpting method enables the removal of occluding material to reveal cylindrical structures of interest intended for such deformation. Finally, we present a set of created visualizations that demonstrates the flexibility of our approach and effectively mimics the form of visualization observed in motivating illustrations.

Keywords Illustrative visualization · Focus+context visualization · Multilevel visualization hierarchy · Volume deformation · Constrained volume sculpting · Pathfinding

Supported by the *Natural Sciences and Engineering Research Council (NSERC)* of Canada, *Alberta Innovates – Technology Futures (AITF)*, *Alberta Enterprise and Advanced Education*, and *Network of Centres of Excellence (NCE)* of Canada in *Graphics, Animation and New Media (GRAND)*.

J. F. Packer · M. Hasan (✉) · F. F. Samavati
Department of Computer Science
University of Calgary, Calgary, Alberta, Canada
e-mail: mhasan@ucalgary.ca

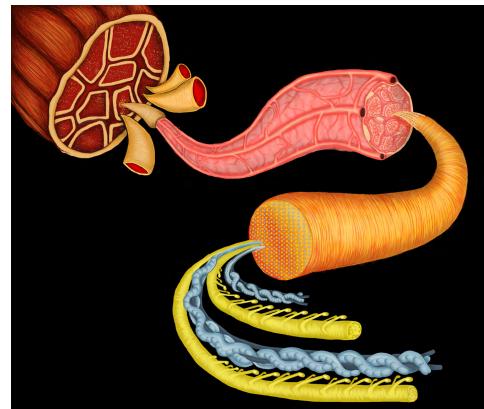


Fig. 1 An illustration motivating the work in this article. A human limb is cut revealing a cross section of the muscle (upper left). The structures that compose its interior are illustrated along a snaking path. This multiscale image demonstrates all structures of muscle tissue down to the microscopic level. (Illustration courtesy of Natasha Shevchenko. Used with permission.)

Mathematics Subject Classification (2000) 65D18 · 68U05 · 76M27

1 Introduction

Context and motivation. Artistic medical illustrations often depict structures in such a way that emphasizes certain features while revealing their nature as a whole. For example, Fig. 1 shows human muscle tissue drawn to emphasize the structure of its various layers while demonstrating that the tissue as a whole is made up of long strands wrapped together. The work in this article is motivated by such artistic illustrations that depict a multiscale visualization of a cylindrical structure with several levels of focus+context. A focus+context effect is

created with the initial context shown in the background, while each successive scale is brought into focus along a snaking path towards the viewpoint. These snaking paths are restricted to a specified space because they are normally presented along with other materials on a page, such as text and annotations. Along the path, several focal points that provide a magnified view of the focal structure are presented, while the path itself acts as the space for transitions between focal points. As we can observe in Fig. 1, this kind of visualization offers more artistic transitions than traditional lens-based (such as [33,34]) and region-based (such as [24,25, 42]) magnification methods, while maintaining clarity and visual continuity.

The multiscale nature of the motivating illustrations demonstrates an important application of creating such visualizations. For example, Fig. 1 shows several resolutions of data ranging from the size of a human arm (centimetres) down to the very microscopic structures that build the muscle in the arm (micrometres). These smallest structures are four orders of magnitude smaller than the size of the largest structure. A volumetric dataset containing every detail on this smallest order of magnitude would be too large to fully load into memory. Splitting the visualization into sections along the path allows loading only portions of each volume, each capturing a different level of resolution. This enables such visualizations to be extremely multiscale.

Goals. The work in this article aims to provide an algorithmic method for the computer generation of artistic multilevel focus+context visualizations similar to the one in Fig. 1 from acquired volumetric data. Utilizing existing acquired volumetric data along with an algorithmic approach for generating the desired illustration enables those with little modeling experience to create such visualizations. Volume data are typically difficult to interact with and deform, thus, our approach to the deformation of volume data should be automatic. Volume data, however, can lack the level of resolution necessary to differentiate between very small structures. Thus, we additionally consider generating a preliminary geometric model along the snaking path that provides a template for further augmentation with additional details using external 3D modeling software.

The goals of this work are based on some key observations that we make about these type of illustrations. The first goal is to algorithmically create a 3D snaking path whose 2D projection is loosely bound in a 2D area. In the motivating illustrations, such as the one provided in Fig. 1, we observe that the path sweeps back and forth across a 2D area, producing its snaking behaviour. This helps depict the structures along the transitions as well as at the focal points. The next goal

is to enable the creation of a model along the snaking path with multiple levels of focus+context, preferably from original acquired volume data. We observe that in the motivating illustrations, several scales are presented along a path. Each scale has its own designated section of the path and these sections end with an enlarged cap, slightly oriented towards the viewpoint to create an area of focus. When combined, multiple levels of focus+context are created.

Problems and challenges. Starting with a volumetric dataset and a 2D illustration space, there are several issues that we must solve in order to create such visualizations. These include: (1) locating the structure of interest in a volume model, (2) creation of a 2D snaking path, (3) generation of a 3D curve whose projection into 2D matches the 2D snaking path, and (4) creation of a model along the snaking curve from the original volume model.

We require a method to interactively explore a volume model so that the structure of interest for the final visualization can be discovered and a visualization of the context can be created. For example, if a major artery in a volume model of a human torso is to be the focus, then the artery must be located in the original model to create a context for the focus.

We separate the creation of a snaking path in 2D from the generation of a 3D snaking curve because the extra dimension introduces additional complexity. If the snaking path were created directly in 3D, it would be necessary to consider the shape of the curve and how it travels through the third dimension simultaneously. Even with the separation of the two problems, the additional third dimension may still be assigned arbitrarily, making it difficult to automatically create a 3D curve from 2D points.

Contributions and methodology. This article provides a flexible approach for generating multilevel focus+context visualizations and models that present focal structures along snaking paths. The paths the structures follow enable the structures themselves to become the transitions between foci and their respective contexts. The paths help to provide clear visual continuity between multiple focal points or scales.

First, we adapt an optimization algorithm to automatically find a 2D snaking path through a specified area. Next, we provide an approach for creating a 3D curve whose 2D projection closely matches the 2D snaking path. Then we create volumetric models along the created 3D curve using acquired volume data, via a shape matching deformation. Geometric models may also be generated to ensure maximum flexibility in the approach. We also adapt the point radiation method of

Chen et al. [7] to create a constrained volume sculpting technique. This technique enables easy removal of occluding material, which aids in discovering structures of interest in volumes for focus+context visualizations.

Article roadmap. The layout of this article is as follows. Sect. 2 presents a short survey of the related work, which is followed by an overview of our general approach in Sect. 3. Sect. 4 describes our method for generating 3D snaking curves and Sect. 5 presents our process for extracting a particular structure from segmented volumetric data for deformation along the generated 3D snaking curve. Next, we describe a method for constrained volume sculpting in Sect. 6. In Sect. 7, we present a comparative evaluation of our visualization technique along with identifying some of its limitations. Finally, Sect. 9 concludes the article with directions for future work.

2 Previous work

Our work falls in the areas of illustrative and focus+context visualization. However, our entire approach utilizes several areas where much previous work has been completed. In this section, we review previous work on focus+context visualization, pathfinding, volume deformation, and volume sculpting – all of which form the basis for our approach.

2.1 Focus+context visualization

Focus+context involves visualizations that reveal a structure of interest in high detail (focus), while demonstrating the origin or location of that structure in an image (context) showing the entire or a larger portion of the host object. Cohen and Brodlie [9, 10] identify two major categories for focus+context techniques: *distorted* and *undistorted*, based on whether any distortion is introduced in the transition from context to focus. Another important distinction that they make is between methods that are *continuous* and that are *discontinuous*, based on how the connection between the focus and the context is made. In this section, we review the related techniques while classifying them based on how the structure of interest is manipulated to direct viewers' attention.

2.1.1 Magnification

Some works in focus+context deal with the magnification of specified portions of the data by placing a lens in the region of interest while preserving the original

context. Carpendale and Montagnese [5] manipulate points on a 2D plane in 3D space according to a lens-based model. They provide different types of transitions ranging from continuous to discontinuous depending on the model of lens used. Other lens-based methods utilize interpolation of data around the edges of the lens to provide continuous transitions between magnified and unmagnified portions [33, 34]. Fisheye views are used in the magic lens for volume data [45], which distorts the original volume around the lens. Distortion magnification methods are usually continuous because they provide transitions between the focus and context via distortion. The focus+context illustration technique we are attempting to mimic in this work can be classified as distortion-based and continuous. The transition areas add information to the focus+context illustrations. A problem with the discussed magnification methods is that the transition areas are often very distorted and would be difficult to modify to portray multiple levels of focus or magnification as the distortions would compound. Illustrating a structure along a snaking path, as in this article, has the benefit of mitigating distortion while enabling multiple levels of clear focus+context.

As opposed to lens-based techniques, region-based techniques are usually undistorted and discontinuous, such as contextual close-ups based on multiresolution [24, 25, 42]. In Sect. 8, we compare our results with that of a region-based technique [24, 25] that also produces *multilevel* focus+context visualizations.

Magnification-based focus+context visualizations are usually *multiscale* because focus and context are rendered at different scales, showing different levels of details. Data for different scales can be obtained through a variety of methods, for example, using a multiresolution technique from the original high-resolution image (as in [24, 25]), from separate image datasets (that may be multi-modal) acquired at different scales (as in [38]), and using non-linear projections (as in [28]). Our focus+context illustration technique is also multiscale because structures are illustrated from a scale demonstrating their original size down to their microscopic interiors. Further comparison of our technique with a continuous multiscale focus+context visualization technique [28] is provided in Sect. 8.

2.1.2 Structural highlighting

Highlighting important structures in volume visualization or illustration is an effective method of directing the attention of viewers. Several methods can be used to highlight important structures within the original context including ghosting and cutaways. In [3], Bruckner et al. present a context-preserving volume rendering

process that utilizes ghosting and several contrast enhancements to shift focus to interior structures. This context-preserving volume rendering process builds on the idea of two-level volume rendering [27], where a focus+context effect is created by rendering the interior layers of the same volume using a different method than is used to render the exterior layers. Examples of other contrast enhancements include contour enhancements [15], as well as depth and focus cues, orientation cues, distance fading, and sketch lines as presented by Ebert and Rheingans [17]. These undistorted techniques do not relocate the focal structure and instead leave them at their original location in the volume. In contrast, the illustrations in our work highlight the long, cylindrical shape of the structures through distortion, which in turn acts as a method of structural highlighting.

2.2 Pathfinding

Pathfinding has received particular attention in robotics and artificial intelligence, however, our purpose for pathfinding is to find a path that fills a certain area. Some pathfinding algorithms reduce to shortest-path algorithms, such as the A* algorithm [23], an improved version of Dijkstra's algorithm [16]. These algorithms require edge costs independent of the path travelled. Our approach for pathfinding deals with the minimum cost Hamiltonian path based on the turning angles between a set of nodes that fill a designated area. Therefore, the cost between two nodes depends on the path travelled up to the current node. Finding a Hamiltonian path based on turning angles can be equated to the angular-metric travelling salesman problem [2], which we use to find a snaking path.

2.3 Volume deformation

Works on free-form deformation (FFD) of volumes include deformation based on a tensor product of trivariate patches by Sederberg and Parry [40]. Chen et al. [6] expand on the original FFD, accelerating it for ray casting in the parallel environment of special purpose graphics hardware. Other works involving the use of special purpose graphics hardware take an object space approach [36] and texture space approach [47].

The previous FFD methods perform global deformation not aiding local exploration. Deformations have been presented as an effective method for browsing volumetric data by McGuffin et al. [32], who introduced local deformation tools such as the ‘leafier’ and ‘spreader’. Utilizing the paradigm of deformation for browsing, Correa et al. [13] present a local deformation method that is

feature-aligned instead of axis-aligned. They provide tools to retract and to cut volumes aligned against either surfaces or segments. In another work, the same authors generalize the idea of displacement maps to volume deformation and provide a process for high-quality modeling and rendering of discontinuous deformations such as peeling [14].

The previously described methods are useful for general deformation and browsing. However, because we only deal with deforming tubular structures, we are able to utilize curve skeletons and create a more specific deformation process. Curve skeletons are important in our deformation process because they aid in providing a mapping from the arbitrary path of the original structure to the desired snaking path. Our approach is similar to the skeleton tree method for animating volumetric models [19], however, we do not grow spheres around skeletal points to reconstruct the deformed volume. Curve skeletons can be found in a variety of ways, such as medial-axis thinning (MAT) based on decision trees [31] and potential field methods [8, 12, 21]. In our approach, we use MAT based on decision trees due to its speed.

2.4 Volume sculpting

Volume sculpting aids in the removal of occluding materials to reveal the structure of interest that will be deformed for the creation of visualizations using our snaking curve approach. It also provides a way to further explore the interior of the volumetric portions of the generated visualizations. Sculpting for volume data was popularized as a way to shape clay or wax using a variety of tools such as ‘sandpaper’ or ‘heat gun’ [20]. These tools were imprecise and highly aliased due to directly calculating tool-volume intersections with a lack of filtering. In [46], Wang and Kaufman introduced a set of carving tools based on shapes of real sculpting tools. Instead of simply determining tool-volume intersections, their tools prevent aliasing by positioning a filter kernel over a sample point and splatting the points of the tool in 3D space onto neighboring sample points. The filters attenuate to a null effect at the radius of the kernel, thereby providing a mechanism of anti-aliasing.

Improvements to the sculpting metaphor include higher quality tools based on potential fields [18] and point radiation [7]. In [18], Ferley et al. define a tool by its potential field and the way that potential field combines with the existing 3D model’s potential field. However, because the tools they create are 3D, they can be awkward to use with a 2D pointing device such as a mouse or stylus. Chen et al. present a point-based

method for an efficient, high-quality sculpting and segmentation based on energy radiating from tool-affected voxels [7].

A method that attempts to constrain sculpting, more for accuracy than for convenience, constrains the depth drilled based on distance gradient changes that occur across successive drilling operations [29]. Anytime the distance gradient change exceeds a threshold the drilling operation is ignored, thus, preventing drilling to an unintended depth. An advantage of this method is that it does not rely on structural information but they provide no way of strictly constraining the drilling to certain structures, such as bones, while keeping the neighboring structures intact.

A problem with any of the previous methods is that they do not differentiate between voxels of different structures. If segmented volume data is available, it is useful to be able to sculpt without the worry of affecting a structure of interest (or, perhaps, affecting only the structure of interest). As part of this article, we constrain the point radiation method of Chen et al. [7] to only operate on a specified range of labels in segmented volume data. This enables easy removal of occluding material, which is necessary to locate structures of interest for deformation along snaking paths.

3 Overview

Our general approach is to algorithmically generate all components for a final visualization based on a few constraints. A flow diagram for our entire process is shown in Fig. 2.

For locating the structure of interest in acquired volume data to be used in the visualizations, we use sculpting operations including chiseling, drilling, and peeling for removing the occluding material.

The process of generating the snaking curve begins by defining a bounding area in which the final space-filling snaking path will be located. As shown in Fig. 2, our approach involves creating a set of points, or nodes, within the space and applying a pathfinding algorithm through these nodes. However, not just any path through the nodes is a desirable path. Observing the motivating illustration in Fig. 1, we see that the shapes or curvatures of the turns are what make the snaking curves desirable.

Thus, we define curves by minimizing their total turning angle while going through all of the 2D nodes. This has the effect of forcing the curve to sweep out as much vertical or horizontal space in the bounding area as possible before turning. We take this algorithmic approach instead of directly creating or drawing a 2D curve (either through sketching or existing control point-based methods such as Bezier curves or B-Spline curves)

because we think of the final visualizations in our work as being user-guided, computer-generated visualizations, where the user-guidance is in the specification of the constraints for pathfinding. Rather than allowing the user to directly create the 2D curve, our algorithmic approach suggests an initial curve path, which then can and typically should be adjusted by the user (even after an illustration has been created) in order to be able to fine-tune the message of the illustration.

A 3D curve must be created once a 2D path is found. Enabling this transformation requires a method for assigning a depth to the position of each 2D node. The resulting 3D points are then used as control points for a curve modeling scheme for generating the snaking curve.

Finally, the structure of interest is deformed to match the shape of the 3D snaking curve. Different sections of the snaking curve will depict different scales. As such, we represent each scale as different resolutions of volume data. The first scale, furthest from the viewpoint, deforms a low resolution version of the volume. The sections of the snaking curve closer to the viewpoint deform gradually higher resolution versions of the volume. We obtain low resolution versions of an acquired volume dataset by repeatedly downsampling it by half as required.

4 Snaking path generation

In this section, we discuss our approach of automatically creating a 3D snaking path given only a few 2D constraints. By observing the shape and properties of snaking paths from a number of motivating illustrations similar to the one in Fig. 1, we have produced the following list of requirements for the path we expect to create.

Utilize a specified 2D space with a snaking path. The path should utilize as much of a specified 2D bounded space as possible, travelling over its entirety, in a snaking path. The snaking path travels through space towards the viewpoint to create the focus. This is how the exterior of each inner layer is demonstrated in the motivating illustrations.

No self-intersection or self-occlusion. If the curve were to self-intersect, then parts of the structures would be occluded at the points of intersection. Any occlusion would hide parts of the exterior of the structure being illustrated along the path and defeat the purpose of the expected illustration.

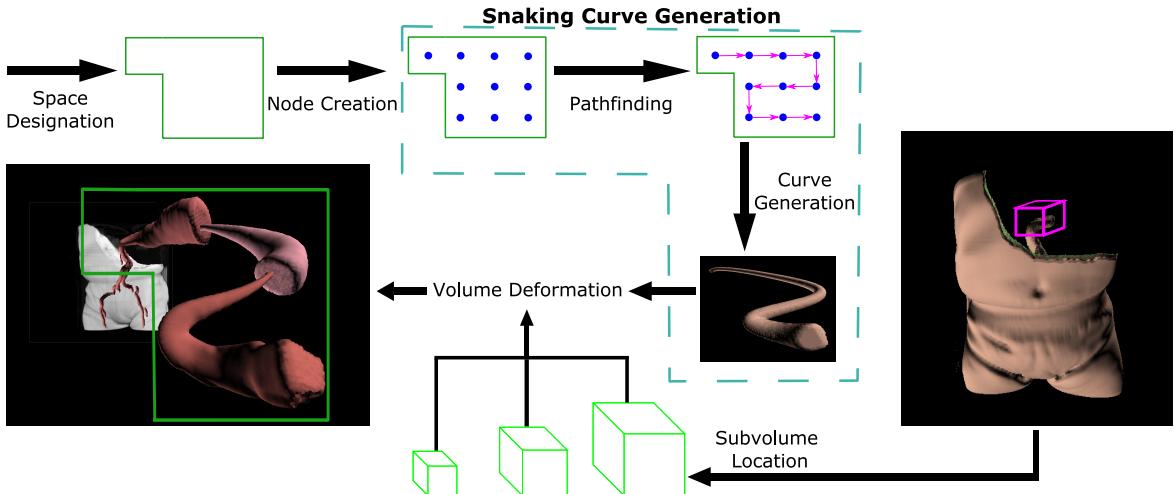


Fig. 2 An overview of the flow of our entire approach used to create multilevel focus+context visualizations along snaking paths. A bounding area is designated to confine the snaking path. Our path generation approach creates a set of points, or nodes, and uses a pathfinding algorithm to generate a path through the set of nodes. A final 3D snaking curve is generated from the 2D nodes in the path using an automatic depth assignment. Finally, the structure of interest is deformed to follow the generated snaking curve. Several resolutions of data can be used to create the multiple levels of focus+context, each containing more detail than the prior level. This flow shows a final visualization involving multiple levels of focus+context of a human aorta.

Tangents. The tangents at both ends of the curve should be customizable. This property helps us to control the view of the structure near the ends of the curve. The construction of our automatic approach is such that tangents can be specified once a path through space is found.

The following three subsections describe our snaking path generation process.

4.1 Pathfinding: ordering 2D points

Similar to the cells in space-filling curve construction (such as [4]), our approach breaks the bounding space for the path into several sections via tessellation. To start with, we consider simple rectangular bounding regions. However, our approach allows more complicated

structures to be built from piecing rectangular regions together. The bounding region is tessellated to produce a regular grid of rectangular cells. Given a grid of cells, we create the shape of the described paths by connecting nodes at the center of each grid cell. Fig. 3 shows two such paths in 2D.

Once we have a grid of nodes, a connectivity graph is created such that every node is connected to its neighbors in the grid. Our goal is to find a Hamiltonian path through the graph that has the minimum total turning angle. This helps to create curves with low curvatures utilizing the available space.

To do this, we define the turning cost between a pair of edges u and v as the angle between the two edges, $\theta_{u,v}$. Over a path P , we seek to minimize $\sum_{u,v \in P} \theta_{u,v}$ for all consecutive edge pairs (u, v) in P . For a fully connected graph, this problem is the angular-metric travelling salesman problem [2].

In our implementation, we interactively define the start and end points for the final curve. We can also specify that the pathfinding disregards some grid nodes, supporting further variations in the resulting paths.

4.2 3D point creation

The path generation in Sect. 4.1 is restricted to two dimensions but the final curve must be in 3D. We solve the problem in two steps. First, we find a way to reverse-project the points into 3D and then we determine a

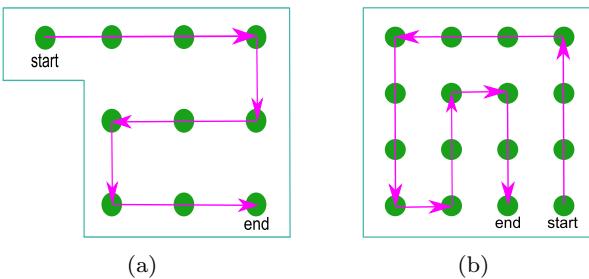


Fig. 3 Two examples of snaking paths. A sinusoidal shape is shown in (a) while a less obvious snaking path is shown in (b).

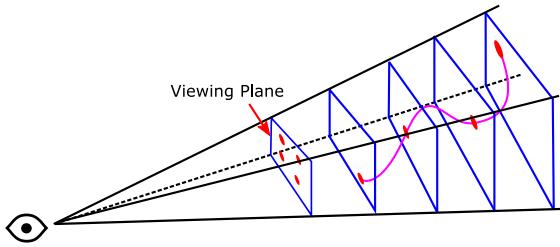


Fig. 4 Rays (not shown) originating from the 2D coordinates in the viewing plane are shot into the scene and intersected with view-aligned planes to find 3D positions. An interpolating curve is formed by utilizing the 3D points.

method for assigning depths to the points via experimentation.

Reverse-projecting the points from 2D to 3D is a typical problem encountered during object selection and the solution is known as a picking operation [44]. We use the same concept to convert each 2D node to a 3D point. Fig. 4 shows this process and an example of a resulting curve.

The next issue is the distribution of the points over the total depth change. Natural choices for spacing the points include those that are easily represented mathematically and are predictable. For example, both uniform and logarithmic spacing of points are supported by our implementation.

4.3 3D snaking curve generation

The 3D points found by the pathfinding technique described in Sect. 4.2 are then used as control points for a NURBS curve. NURBS is chosen because NURBS allows weighted control points. Weighted control points are used as an additional control over the shape of the curve without moving the control points. The specified start and end tangents are also taken into consideration at this step.

5 Creating models for the snaking paths

In this section, we discuss how to create a model to fit the snaking curve we generated in Sect. 4. We describe a process for deforming existing volume data to follow the generated path. We also demonstrate that it is possible to combine the initially rendered volume with geometric models for depicting certain portions with extra details. Such extra details may be introduced either because they are absent in the associated volume data or for a more artistic effect.

The goal is to visualize different structures along the snaking curve. Thus, it is necessary to somehow segment

the volumetric data. The segmentation can be performed using any method such as level set segmentation [37] or a flood fill approach [7]. However, the method of segmentation does not matter for our approach as it only requires that a classification of each voxel exists.

Assuming separate structures have been segmented and labelled uniquely, we perform a deformation of the structure of interest into the shape of the 3D snaking curve. For this, a mapping from the shape of the structure to the shape of the curve is created. However, such a mapping can be difficult to find. The process to find this mapping and performing the deformation is described in Sect. 5.1.

Part of mimicking the motivational illustrations includes enabling the visualization of several focal areas along the generated snaking curve. As such, we break the generated snaking curve into several sections, each terminating with a focal area. This enables the visualization of multiple levels of focus+context within the same illustration. Sect. 5.2 describes the approach taken in this article to enable the generation of multilevel focus+context visualizations.

5.1 Volume deformation

Deformation of the volumetric structure of interest is performed by mapping its curve skeleton to the shape of the snaking curve generated for the expected visualization. We take a sweep volume approach to complete the deformation due to the generalized cylindrical nature of the structure. A graphical outline of the steps for the deformation is shown in Fig. 5.

A sweep volume approach involves sweeping a template along the trajectory of a curve. The contour of the template defines the outer surface of the sweep volume as the template is swept along the trajectory of the curve. We set out requirements on the structure to be deformed and describe how we map the shape of the structure to the snaking curve for the deformation.

5.1.1 Focal structure requirements

In this work, the structures that are specified for use in the deformation should have two properties – they should be cylindrical and branchless. They should be cylindrical because of the nature of the motivating illustrations. That is, the type of illustration being mimicked are created for depicting the interior layers and substructures of cylindrical objects. The structures should be branchless because in this work, the snaking curve onto which a structure is mapped is branchless. Combining

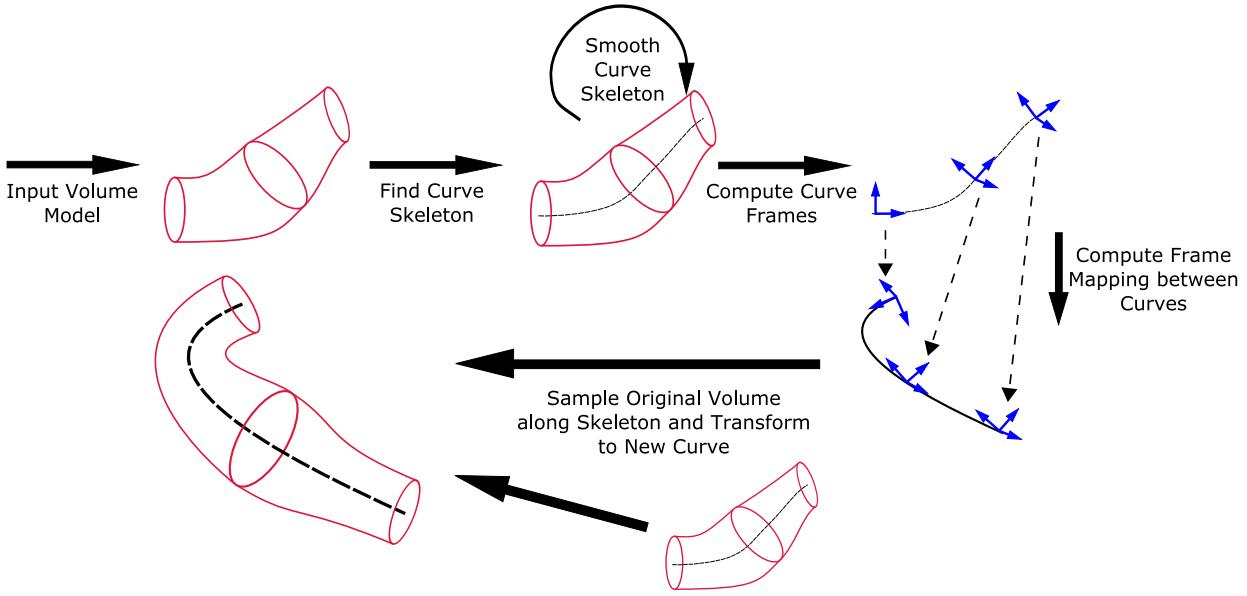


Fig. 5 An outline of the deformation process

these requirements results in a general shape that we assume the structures of interest used in our visualizations will have – a generalized cylinder.

5.1.2 Focal subvolume creation

In order to perform the deformation efficiently, we select only a subvolume of the original volume containing the focal structure for deformation. The subvolume is selected using an axis-aligned bounding box. The final focal subvolume is created by masking the original volume with a unique integer label for the designated structure (given through the segmentation data) within the selected subvolume.

5.1.3 Curve skeleton extraction

A curve skeleton of a 3D shape is a “simplified 1D representation of the medial surface” [11]. For speedy extraction of curve skeletons, we use the medial axis thinning (MAT) method of Lee et al. [31] that relies on decision trees. Thinning methods for finding the curve skeleton involve eroding the contour of an image until a single pixel/voxel wide curve skeleton is discovered.

The curve skeleton resulting from MAT is generally noisy because thinning is sensitive to surface perturbations [11]. This jagged curve skeleton is not ideal for the mapping and deformation that are later performed because the local frames (see Sect. 5.1.4) will replicate the jagged nature and cause a discontinuous mapping for the deformation. Therefore, a smooth curve is created from the resulting jagged curve skeleton by

applying cubic B-spline reverse subdivision [39]. The smooth curve is a good approximation of the curve skeleton unless too many levels of reverse subdivision are applied. In our experience, three levels of reverse subdivision create a smoothed curve skeleton without adversely affecting the overall shape. Utilizing a potential field approach [8, 12, 21] for finding the curve skeleton may remove the need for smoothing, and thus, a change of shape of the curve skeleton from smoothing would no longer be a factor.

5.1.4 Local curve frames mapping

The sweeping deformation requires a method of traversing both the curve skeleton and the snaking curve. At each point along each curve, the sweeping template must be positioned and oriented appropriately. For this, we utilize local frames that represent position and orientation computed from the curves. Finally, a mapping between the frames of the curve skeleton and that of the 3D snaking curve must be found to define the transformation between the two curves.

In order to reduce drastic frame changes, we use parallel transport frames [22] for calculating the local frames. Finding parallel transport frames involves using the tangent vectors for the curve and an initial normal vector [22]. The tangent vectors define the ‘forward’ vectors for each frame while the normal vectors define the ‘right’ vectors (which point to a curve that is parallel to the original curve, hence the term ‘parallel’ transport frames).

Algorithm 1 Our approach to generating a 3D model for the snaking path.

Require: The structure of interest and the generated snaking curve.

- 1: Extract the curve skeleton for the structure of interest using MAT as described in Sect. 5.1.3.
- 2: Apply cubic B-spline reverse subdivision [39] to the resulting curve skeleton to smooth its jaggedness.
- 3: Create a cubic B-spline curve from the points that remain after the reverse subdivision in step 2. This is the curve skeleton that will be used to find the local frames.
- 4: Compute and store the parallel transport frames of the curve skeleton created in step 3.
- 5: Map points on the curve skeleton found in step 4 to points along the snaking curve with respect to arc length.
- 6: Use the mapping created in step 5 to transform the frames of the curve skeleton into the frames of the generated snaking curve. Thus, this also requires computing the parallel transport frames for the snaking curve.
- 7: Sample the original subvolume (structure of interest) along the trajectory of the curve skeleton. Transform the position of the samples according to the mapping and transformation obtained in step 6.
- 8: Resample the deformed samples onto a uniform grid because they are currently scattered data.
- 9: **return** The structure of interest deformed along the snaking curve.

The mapping between the curve skeleton and the generated snaking curve is performed by mapping points between the start and end of the curve skeleton using linear interpolation. Points are mapped along the two curves with respect to arc length.

5.1.5 Sweeping deformation

In the sweeping deformation, a template for sampling the original volume is swept along the trajectory of the curve skeleton of the structure of interest. As the template is swept, the original volume is sampled and the positions of the samples are transformed to a corresponding position and orientation on the snaking curve using the curve frame mapping described in Sect. 5.1.4. The contour of the template, defining the outer surface of the deformed volume, is scaled during the sweep. This scaling helps in creating the effect of gradual magnification of volume substructures during the transitions from contexts to foci.

The steps for the entire deformation, shown pictorially in Fig. 5, are given in Algorithm 1. Steps 1 through 4 involve processes that were discussed in the previous sections. Thus, the rest of the steps are now described in more detail. Steps 6 and 7 involve iterating over each frame in the trajectory of the curve skeleton and using the mapping onto a frame of the generated snaking curve from step 5. At each location along the curve skeleton, a sweep template is oriented according to the

frame at that location and is used to sample the original volume. The positions of the samples in the original subvolume are transformed according to the transformation between the current frame in the curve skeleton and its corresponding frame in the snaking curve. The sample values and their new positions are stored and are used to reconstruct the deformed volume.

After the mapping of the original samples into their new positions along the generated snaking curve, resampling onto a uniform grid is required in order to visualize the result utilizing volume rendering via texture mapping. For this purpose, we use scattered data interpolation via the inverse distance weighting of Shepard's method [41], available in VTK [30]. It is critical for the scattered data interpolation that the deformed positions be scattered fairly evenly along the generated snaking curve so that no large empty areas affect the resampling. More local resampling methods may be more appropriate. Examples of local resampling methods include those designed for image-swept volumes in which resampling occurs after the mapping of each sweep segment [48] and for resampling volumes around skeletal points by growing spheres for the sampling areas [19], which are scan filled.

5.2 Multilevel focus+context

The long, unobscured snaking path can be used for multiple areas of magnification or focal points, each terminating a portion of the snaking path. As we observe in the motivating illustration in Fig. 1, each section of the snaking path depicts structures at a different scale. The sections farther away from the viewpoint show structures on a small scale while the sections closer to the viewing plane show structures on a very large scale. To mimic this effect, the generated snaking path is broken into sections. Each section is used for a different scale or focal level. In a sense, each previous focal level acts as the context for the next level. This allows a final visualization utilizing the generated snaking curve to contain multiple levels of focus+context with a smooth transition between each level.

During volume deformation along the snaking path, volume data from successively higher resolutions of the same dataset are used in each section of the path, starting from the section farthest away from the viewpoint to the section closest to the viewing plane. If the final curve only has one section, then only one level of resolution is used.

There needs to be a mapping between resolutions for locating the structure of interest in each resolution. The curve skeleton for the structure of interest is then found in each resolution separately. The sweep deformation

process sweeps along the curve skeleton of the lowest resolution until the end of the first section of the snaking curve is reached. The voxel location of the curve skeleton at this point is then mapped into a voxel location in the successive higher resolution and the sweeping deformation continues along the curve skeleton in the higher resolution data.

In this work, when different resolutions of acquired volume data were required, we obtained lower resolutions by repeatedly downsampling the original volume data by half. Thus, when locating a voxel along the curve skeleton for the successive higher resolution, the voxel coordinates are multiplied by two. The curve skeletons are used to find the voxel locations to prevent any mapping errors. Once the voxel locations are found in the higher resolution, the nearest location on the smoothed curve skeleton is used as the location to resume the sweeping deformation.

It is also possible to use data obtained from different modalities [such as computed tomography (CT) and magnetic resonance imaging (MRI) data] for different portions of the snaking path instead of only using different resolutions of the same data. The only requirement is to have a mapping between the datasets with different modalities. The mapping will be one-to-one if the datasets are co-registered in the same resolution.

Furthermore, allowing the use of data with different resolutions for different sections of the snaking path, from possibly different datasets, enables the creation of models and visualizations across structures, whose sizes differ in several orders of magnitude. For example, an aorta in the human body is on the scale of centimetres but blood cells are on the scale of micrometres. A dataset of a human torso containing the aorta captured at a resolution small enough to capture the structure of blood cells would be far too large to fully load into memory. More specifically, a volume with 8 bits per voxel of size $15\text{cm} \times 15\text{cm} \times 20\text{cm}$ of the human torso, captured at a resolution such that structures measuring only micrometres in length could be seen, would be approximately 320 petabytes in size. By splitting the visualization into sections along the path, we enable the ability to load only the required portions of each volume. Some of these volumes may be captured at a very large scale while the others may be captured at smaller scales. This enables our approach to be extended to extreme multiscale visualizations.

However, a problem with using acquired volume data (such as CT data used in this work) is that sometimes it does not contain enough information to discern between interior layers. In the case of a human aorta, these could include things such as the layers of the vessel wall, the

blood flowing through the vessel, and at a microscopic level, the blood cells the aorta is transporting.

Our approach additionally allows a surface model to be created to follow the generated snaking path in any designated section, in cases where the volume data are not sufficient to illustrate a structure in a larger scale or with greater detail. We generate a surface model of a generalized cylinder along the generated snaking path to provide an initial model that an artist can later edit or augment with further details using modeling tools such as Blender [1]. This creates a hybrid visualization that mixes the benefits of exploring real acquired volume data with the imagination of an artist to illustrate extra details that the original data may lack.

6 Constrained volume sculpting

In order to create images similar to the motivating illustration shown in Fig. 1, it is necessary to have a method that removes occluding material without affecting the structure of interest. In the motivating illustration, the muscle tissue in a limb provides the context for the snaking curve illustration that shows several focal sections containing different substructures of the muscle tissue. It depicts part of the muscle cut away to reveal a cross section of the muscle tissue. Exterior structures are also removed. However, some smaller interior structures at the point of the cut remain to provide further context. In our illustrations, we aim to mimic the same effect of removing occluding structures from a volume model to provide a clear context for the location of the structure of interest, while leaving it intact.

Sculpting operations that deal with the removal of material, such as chiseling and drilling, are often difficult to control. It is very easy to accidentally remove or destroy parts of a structure that was to remain unaffected. Thus, we aim to constrain these types of sculpting operations so that they are able to preserve structures that should remain unaffected by them.

For this purpose, we constrained the sculpting by point radiation technique of Chen et al. [7]. The point radiation technique enables the creation of high-quality, interactive sculpting tools that are anti-aliased by their nature, making it an attractive choice for sculpting. A set of 3D footprints is rendered to a *sculpting volume*, which contains a collection of radiated points created by tool-volume interactions. It is this collection of radiated points that defines the combined influence of the sculpting operations. The sculpting volume is used to mask the original volume during rendering. While rendering the original volume, the sculpting volume is sampled and if the value contained in the sculpting volume is below a certain threshold, only then the original volume

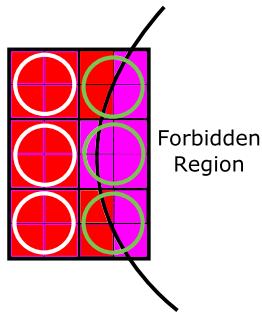


Fig. 6 A simplified 2D illustration of voxels that must be discarded (shaded red) during constrained point radiation. Footprints covering sets of voxels, designated by the circles, and a curve that denotes the surface of a forbidden region that should not be affected by the current interaction are shown. The magenta shaded voxels represent areas that should not be sculpted because they lie within the forbidden region. Notice that some footprints cover areas that should be sculpted as well as areas that should not be sculpted.

value is rendered. This creates the effect of deleting material from the original volume.

Our constrained point radiation is an extension of point radiation that limits the removal of material to certain voxel labels. This creates an additional storage requirement for segmentation data in the form of an integer *label volume*. The label volume contains per-voxel labels that classify the voxels of the original volume. The label volume is stored as an additional 3D texture that is sampled when it is necessary to classify voxels.

We call the region of a volume that should not be affected by the current sculpting operation the forbidden region. In the point radiation technique, the 3D Gaussian value for each footprint determines the value to write to the sculpting volume. We use an additional test to exclude voxels that lie within the forbidden region from the sculpting volume.

However, all 3D footprints within a forbidden region cannot be fully excluded from the sculpting. The problem is that some footprints contain areas that lie within the forbidden region as well as areas that lie within the region to be sculpted, as shown in Fig. 6. The footprints in the right column in Fig. 6 are contained within the forbidden region but if they are completely excluded from the sculpting then the surface resulting from the sculpting will be incorrect. Instead, it is necessary to determine which voxels within each footprint are in which region. When considering voxels in the footprints for storage into the sculpting volume, the label volume is sampled at the same voxel coordinates to see if the label matches a label designating the forbidden region. In case of a match, a voxel remains unaffected by the current sculpting action.

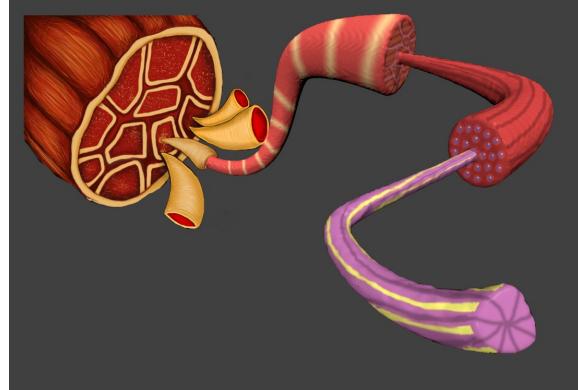


Fig. 7 A visualization of different substructures of muscle tissue created in likeness to the motivating illustration in Fig. 1. The 3D model along the snaking curve was textured in Blender [1] after it was generated using our approach. The rendering is overlaid onto the muscle from the motivating illustration in Fig. 1 for context.

It is important to note that the integer label volume is sampled using the nearest neighbor sampling to preserve the integer label. If the values were sampled using a method of interpolation, it would be possible to retrieve invalid or unexpected labels. Using nearest neighbor sampling, however, has the side-effect of creating aliasing around structures that have all occluding materials removed. This can be addressed by an existing method by Tiede et al. that deals with rendering integer labelled volumes in high quality [43]. Their method mitigates the inability to interpolate between the labels by counting the number of labels surrounding a particular location. If an above average or median amount of labels for one particular structure lie in the neighborhood of the currently considered position, then the currently considered position is assumed to be in the particular structure. Such a rendering method is required for our constrained sculpted data.

7 Results

The presentation of results from our approach and the associated discussion are split into several sections. We present a multilevel focus+context visualization created from texturing a result from the curve generation process (without the use of any acquired data) in Sect. 7.1. This provides a proof of concept for evaluating the effectiveness of our generated snaking curves in the desired multilevel focus+context style. Then in Sect. 7.2, in order to demonstrate the versatility of our approach, we present several visualizations using image-swept volume [48] data, using acquired volumetric data, and using our hybrid volumetric-geometric approach.

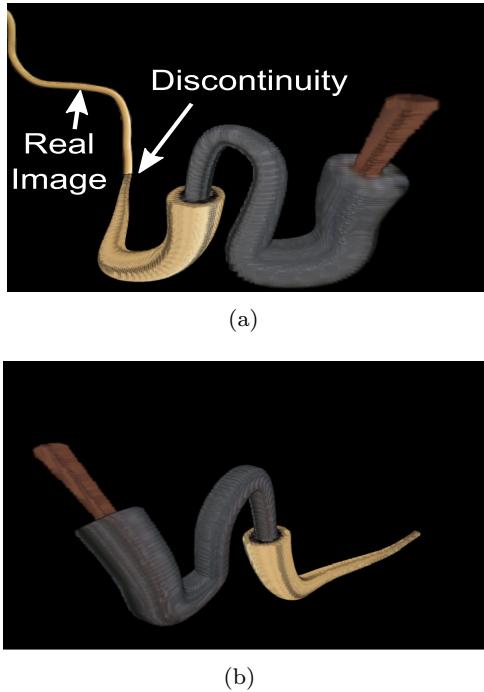


Fig. 8 A visualization of a coaxial cable produced using our approach. The original visualization with an image of a real coaxial cable is shown (a). A rotated view of the visualization is shown in (b), without the image of a real coaxial cable.

7.1 A visualization from a textured snaking model

Although a portion of this article involves utilizing acquired volume data to create the intended visualizations, we have created a visualization similar to a motivating illustration to demonstrate that it is possible to use only the curves themselves to create meaningful multilevel focus+context visualizations along the snaking curves.

An example of a visualization created in the likeness of the motivating illustration in Fig. 1 is shown in Fig. 7. A volumetric model of a generalized cylinder was used to create the 3D model that follows the generated snaking curve. Extra editing in the form of coloring or texturing the resulting model is required to produce informative visualizations because the generalized cylinder contains no information about any structure that would be present in volume data. However, an artist can use the generalized cylinder as a template for the path of the structure through space and create a model with more detail or add texture, which is demonstrated here.

7.2 Generated visualizations

The first three visualizations in this section, shown in Figs. 8, 9, and 10, are based on image-swept volumes [48], produced by sweeping image templates along the tra-

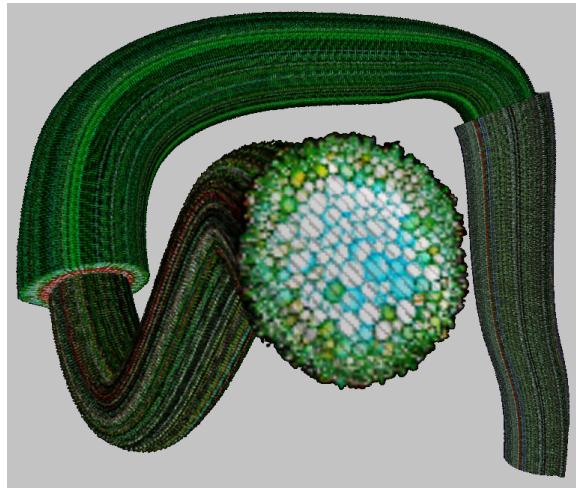


Fig. 9 A visualization from a magnified, stained cross section of a *Hypericum perforatum* plant stem. The snaking curve for this visualization was produced using the setup shown in Fig. 3(b) with uniformly spaced control points in depth. (Data source: Rolf-Dieter Mueller. *Hypericum perforatum* (St. John's Wort) stem cross section. Used under the Creative Commons Attribution 3.0 Unported license.)

jectory of the snaking curve. The image templates used for these three visualizations contain cross sections of a coaxial cable, a plant stem, and a prehistoric fossilized tree trunk, respectively.

A visualization of a coaxial cable created from an image of its cross section we captured is shown in Fig. 8 from two different viewpoints. Fig. 8(a) shows the original view of the visualization overlaid onto a background containing an image of a real coaxial cable, while Fig. 8(b) shows the same generated visualization rotated to show the side of the model, without the background image. Three different sections of the cable are shown. From the left to the right side of the image, these are the exterior insulation, the interior insulation, and the copper wire. Each layer has its exterior portrayed along the snaking curve and the focal cross sections demonstrate the size of each successive layer compared to the previous one. The side view of the coaxial cable shown in Fig. 8(b) reveals a noticeable change in how the curve travels through space near the copper wire. This is due to the specified end tangent to force the copper portion to be oriented more towards the viewpoint.

Fig. 9 shows a visualization created from the image of a stained cross section of a *Hypericum perforatum* plant stem. This visualization uses the example 2D path as shown in Fig. 3(b), demonstrating how that path appears in a final visualization. An illustrative effect is created because a plant stem normally would not bend in such a way.

The visualization in Fig. 10 is created from the image of a cross section of the fossil of a *Hermanophyton*

tree trunk. The goal of this visualization is to bring the inner black portion of the fossil into focus in front of the viewpoint. The stretched, black portion of the fossil appears as though it is ready to slide back into place inside the previous layer. The same could be said for the previous layer. This visualization is successful in demonstrating the effectiveness of multilevel focus+context along snaking paths because it enables the visualization of the exact location of the origin of each layer.

The remaining visualizations in this section, shown in Figs. 11 and 12, utilize acquired volume data for the initial contexts as well as along the generated snaking curves. The approach for volume deformation presented in Sect. 5 is used to deform original volume data along generated snaking curves. All visualizations presented in this section use logarithmic depth-steps for the automatic depth assignment to produce the final 3D curve. Logarithmic spacing is chosen over uniform spacing because logarithmic spacing produces a curve that is oriented more towards the viewpoint along the entire length.

The dataset used to create the visualization presented in Fig. 11 is a pre-segmented, stented, human aorta dataset. The visualization shows three levels of focus+context created from this dataset. A portion of the aorta near the heart was selected as the structure of interest and the occluding materials were removed by our constrained volume sculpting method. The bounding box confining the snaking curve and a curve structure involving two 180° turns (in a three-by-three grid of nodes) were specified for the automatic snaking curve generation. The start tangent was set such that the resulting snaking curve follows the original dataset at its beginning. The end tangent of the snaking curve is a result of logarithmic-based depth steps between control points.



Fig. 10 A visualization of the fossil of a Hermanophyton tree trunk. The original fossil cross section is shown in the lower left corner. (Data source: Cross section of a fossil of a Hermanophyton tree trunk, courtesy of Mike Viney and the Virtual Petrified Wood Museum. Used with permission.)

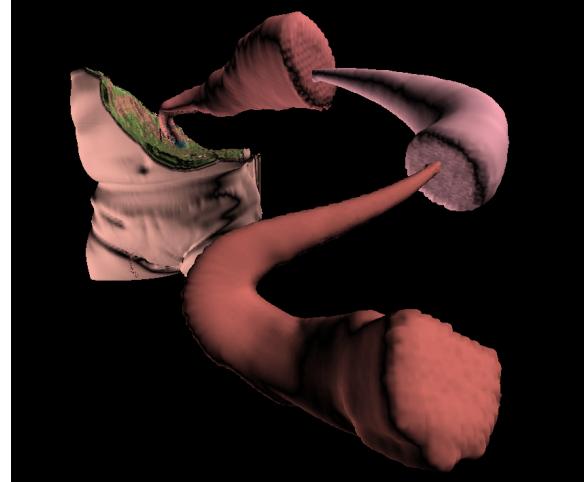


Fig. 11 A visualization containing three sections along the snaking curve. Each section receives its volume data from a consecutively higher resolution of the original volume data. Because the original volume data was already fairly low resolution, the differences are minor. The bulges along some parts of the snaking path are due to bulges on the corresponding parts of the human aorta – the original cylindrical structure before deformation. (Data source: Human aorta dataset, courtesy of Michael Meißner, Viatronix Inc., USA via <http://www.volvis.org>.)

Fig. 12 is generated using our hybrid approach involving both deformed volume data and generated geometric models. It is based on the same dataset as Fig. 11. The volume model of the human torso in the background is of a lower resolution, while the aorta deformed over first half of the snaking curve is of a comparatively higher resolution. Then a surface model is created along the last portion (second half) of the snaking curve. The surface model was exported and edited in Blender [1] to include red blood cells, depicting a microscopic view of oxygenated blood flowing through the aorta.

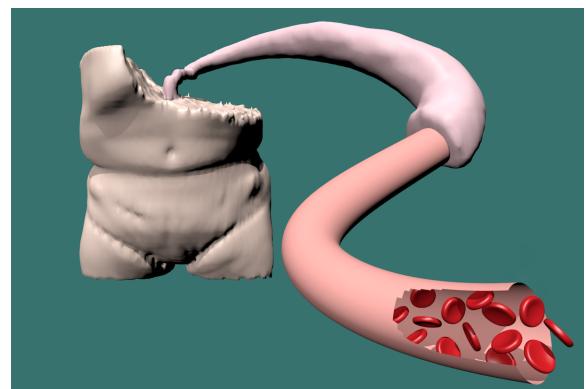


Fig. 12 A visualization using the same human aorta dataset used for Fig. 11 but augmented with geometric models. A surface model is created along the last portion of the snaking curve, closest to the viewpoint. The red blood cells were added afterwards using Blender [1].

Fig. 12, containing the edited surface model proves that if the original volume model does not contain enough detail to differentiate microscopic structures, our approach enables rapid preliminary modeling. Additional details and structures that cannot be captured and illustrated based on volume data can be completed by an artist, using the generated surface model as a guide. This saves the artist the time of creating a model along the entire snaking path and they can instead focus on the creation of the expected structures missing in the volume data.

8 Comparative evaluation and limitations

In this paper, we introduce a novel approach to automatically create multilevel focus+context visualizations of long, layered, cylindrical structures. To the best of our knowledge, this particular problem has not been addressed in the literature prior to this work. However, due to the similarity in resulting visualizations, in this section, we first compare our proposed technique with those of two state-of-the-art visualization approaches. Next, we state the criteria that were taken into consideration for the qualitative evaluation of our results and outline some limitations of our method.

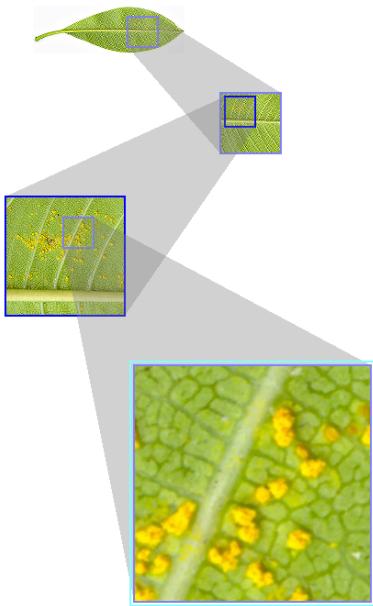


Fig. 13 A multilevel focus+context visualization, where the main context (at the top) remains at one resolution and the magnified focal points are depicted at higher resolutions. Image courtesy of Mahmudul Hasan, generated from the image of a diseased Plumeria rubra leaf with dimensions 10496×3328 , using the technique presented in [24, 25]. (Data source: Sam Fraser-Smith. Frangipani rust (caused by *Coleosporium plumeriae*) on *Plumeria rubra*. Used under the Creative Commons Attribution 2.0 Generic license.)

The first visualization approach we consider for comparison purposes is proposed by Hasan et al. [24, 25]. In their work, they create a multilevel focus+context hierarchy, where a focus may further act as a context (see the visualization in Fig. 13, for example). They support the construction of this multilevel hierarchy using a balanced wavelet transform (BWT) of the original image [26]. Although we do not allow branching structures as they do in their work, we allow the construction of a multilevel focus+context hierarchy with more artistic and illustrative transitions from contexts to foci. Another important distinction is that in our work, the transitions from contexts to foci are constructed from the data being visualized.

The second visualization approach we compare our method with is by Hsu et al. [28]. They present a continuous multiscale focus+context visualization technique that involves shooting non-linearly bent rays from several cameras into a scene. Each of those several cameras captures a particular scale and viewpoint, which are blended to obtain a continuous multiscale visualization. Some of their generated visualizations are similar to ours, however, they emphasize on visualizing multiple scales in a single image with smooth transitions between different scales instead of visualizing one particular structure along a path and lack the ability to interactively manipulate the result in 3D. As such, their approach can only produce images while our approach can produce models that can be further manipulated.

With respect to comparison of the computational costs, the visualization approach proposed by Hasan et al. [24, 25] is interactive except for the part where they construct the BWT in a preprocessing phase. Construction of a BWT takes $O(n)$ time, n being the total number of samples (pixels/voxels) in the image. For the method of Hsu et al. [28], depending on the intended resolution, generation of a desired visualization may take a few minutes to several hours. It is slow because the camera ray generation process is computationally expensive. In contrast, the only part of our system that is not interactive is where we perform scattered data interpolation for volume deformation using the implementation of Shepard's method [41] from VTK [30] (in step 8 of Algorithm 1). Shepard's method exhibits low performance for large datasets with a time complexity of $O(n)$ for interpolation, where n is the number of samples being processed. Modified Shepard's method [35] performs better on large datasets, however, the structure of interest usually contains much less number of samples compared to that of the host dataset.

There are two criteria we used to qualitatively evaluate if a deformed structure of interest is a success. The first one is whether or not it is following a snaking path.

The second and the most important criterion is the shape and orientation of the curve as it travels through space toward the viewpoint. In particular, the tangents along the curve should be oriented mostly towards the viewpoint instead of away from the viewpoint. Curves that are mostly oriented towards the viewpoint are better because there are more spots along the curve where focal areas can be placed.

One limitation of our work is the view-dependent nature of the resulting curve. This is apparent in Fig. 8(b) where, when viewed from a different viewpoint than that used for the generation, the curve appears less desirable and it is not apparent that it is a snaking path. However, recall that our curve generation approach is view-dependent, thus, we do not assume that the snaking curve is visually appealing from other viewpoints. This limits the aesthetics of our approach to static visualizations. Nonetheless, there could still be a benefit in exploring the curve from alternate viewpoints such as for sculpting to further discover the interior structures.

Another limitation is that surface detail is lost due to deformation. The approach works well for objects that have no fine texture or surface structure. However, those with more detailed surface textures would be stretched and deformed, especially if the length of the curve (thus, the amount of deformation) is large. This is mitigated by the use of successively higher resolutions of the structure of interest along each section of the snaking curve as mentioned in Sect. 5.2 but the amount of preserved detail is then limited by the highest resolution.

9 Conclusion and future work

In this article, we have presented an approach to create multilevel, distortion-based, continuous, multiscale focus+context illustrations. These depict a structure of interest travelling along a snaking path, similar to Fig. 1. Our approach enables the automatic creation of 3D snaking curves with a simplified 2D interface. Acquired volume data can be deformed to follow the generated snaking curve, thereby, enabling the creation of focus+context illustrations. When acquired volume data lack the necessary detail, we allow the creation of a geometric model of a generalized cylinder to be created along a portion of the curve, which acts as a template for an artist to build upon.

Our results demonstrate that our approach allows the generation of illustrations in likeness to the motivating illustration in Fig. 1. The resulting visualizations and 3D models can be further explored, by sculpting or viewpoint changes, adding value to our approach.

We found that sometimes volume data do not contain enough detail to achieve results with adequate number

of levels of focus+context. To mitigate this, we added a hybrid approach to create a preliminary geometric model over a portion of the snaking curve. The geometric model then acts as a template for an artist to create a more detailed model. An extension to this is to provide a mechanism for geometric deformation so that our approach can also be used with existing geometric models in the same way it has been used with volume models.

A further expansion to the idea of visualizing structures of interest along snaking paths involves supporting branching structures, which may allow illustrative visualization of more complex structures. Last but not the least, because our approach enables 2D modeling for automatic generation of 3D snaking curves, a natural future work is to integrate our approach into a sketch-based modeling (SBM) system for volumetric data.

References

1. Stichting Blender Foundation. Blender (2012). URL <http://www.blender.org>
2. Aggarwal, A., Coppersmith, D., Khanna, S., Motwani, R., Schieber, B.: The angular-metric traveling salesman problem. *SIAM Journal on Computing* **29**(3), 697–711 (2000). DOI 10.1137/S0097539796312721
3. Bruckner, S., Grimm, S., Kanitsar, A., Gröller, M.E.: Illustrative context-preserving volume rendering. In: Proceedings of the 7th Joint Eurographics/IEEE VGTC Conference on Visualization, EUROVIS, pp. 69–76. Eurographics Association, Aire-la-Ville, Switzerland (2005). DOI 10.2312/VisSym/EuroVis05/069-076
4. Cannon, J.W., Thurston, W.P.: Group invariant Peano curves. *Geom. Topol.* **11**(3), 1315–1355 (2007). DOI 10.2140/gt.2007.11.1315
5. Carpendale, M.S.T., Montagnese, C.: A framework for unifying presentation space. In: Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology, UIST, pp. 61–70. ACM, New York, USA (2001). DOI 10.1145/502348.502358
6. Chen, H., Hesser, J., Männer, R.: Ray casting free-form deformed-volume objects. *J. Visual. Comp. Animat.* **14**(2), 61–72 (2003). DOI 10.1002/vis.305
7. Chen, H.L.J., Samavati, F.F., Sousa, M.C.: GPU-based point radiation for interactive volume sculpting and segmentation. *Vis. Comput.* **24**, 689–698 (2008). DOI 10.1007/s00371-008-0249-5
8. Chuang, J.H., Tsai, C.H., Ko, M.C.: Skeletonisation of three-dimensional object using generalized potential field. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(11), 1241–1251 (2000). DOI 10.1109/34.888709
9. Cohen, M.: Focus and context for volume visualization. Ph.D. thesis (2006)
10. Cohen, M., Brodlie, K.: Focus and context for volume visualization. In: Proceeding of the Theory and Practice of Computer Graphics Conference, pp. 32–39. IEEE (2004). DOI 10.1109/TPCG.2004.1314450
11. Cornea, N., Silver, D., Min, P.: Curve-skeleton applications. In: Proceedings of the Conference on Visualization,

- VIS, pp. 95–102. IEEE Computer Society (2005). DOI 10.1109/VISUAL.2005.1532783
12. Cornea, N.D., Silver, D., Yuan, X., Balasubramanian, R.: Computing hierarchical curve-skeletons of 3D objects. *Vis. Comput.* **21**, 945–955 (2005). DOI 10.1007/s00371-005-0308-0
 13. Correa, C., Silver, D., Chen, M.: Feature aligned volume manipulation for illustration and visualization. *IEEE Trans. Vis. Comput. Graph.* **12**(5), 1069–1076 (2006). DOI 10.1109/TVCG.2006.144
 14. Correa, C.D., Silver, D., Chen, M.: Discontinuous displacement mapping for volume graphics. In: Proceedings of the Eurographics/IEEE VGTC Workshop on Volume Graphics, pp. 9–16. Eurographics Association, Boston, Massachusetts, USA (2006). DOI 10.2312/VG/VG06/009-016
 15. Csébfalvi, B., Mroz, L., Hauser, H., König, A., Gröller, E.: Fast visualization of object contours by non-photorealistic volume rendering. *Comput. Graph. Forum* **20**(3), 452–460 (2001). DOI 10.1111/1467-8659.00538
 16. Dijkstra, E.W.: A note on two problems in connexion with graphs. *Numer. Math.* **1**, 269–271 (1959). DOI 10.1007/BF01386390
 17. Ebert, D., Rheingans, P.: Volume illustration: Non-photorealistic rendering of volume models. In: Proceedings of the Conference on Visualization, VIS, pp. 195–202. IEEE Computer Society Press, Los Alamitos, CA, USA (2000). DOI 10.1109/VISUAL.2000.885694
 18. Ferley, E., Cani, M.P., Gascuel, J.D.: Practical volumetric sculpting. *Vis. Comput.* **16**, 469–480 (2000). DOI 10.1007/PL00007216
 19. Gagvani, N., Kenchammana-Hosekote, D., Silver, D.: Volume animation using the skeleton tree. In: Proceedings of the IEEE Symposium on Volume Visualization, VVS, pp. 47–53. ACM, New York, USA (1998). DOI 10.1145/288126.288152
 20. Galyean, T.A., Hughes, J.F.: Sculpting: An interactive volumetric modeling technique. In: Proceedings of the 18th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH, pp. 267–274. ACM, New York, USA (1991). DOI 10.1145/122718.122747
 21. Grigorishin, T., Abdel-Hamid, G., Yang, Y.: Skeletonisation: An electrostatic field-based approach. *Pattern Anal. Appl.* **1**, 163–177 (1998). DOI 10.1007/BF01259366
 22. Hanson, A.J., Ma, H.: Parallel transport approach to curve framing. Tech. rep., Indiana University (1995)
 23. Hart, P., Nilsson, N., Raphael, B.: A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans. Syst. Sci. Cybernetics* **4**(2), 100–107 (1968). DOI 10.1109/TSSC.1968.300136
 24. Hasan, M., Samavati, F.F., Jacob, C.: Multilevel focus+context visualization using balanced multiresolution. In: International Conference on Cyberworlds (CW) 2014, pp. 145–152. IEEE (2014). DOI 10.1109/CW.2014.28
 25. Hasan, M., Samavati, F.F., Jacob, C.: Interactive multilevel focus+context visualization framework. *The Visual Computer* pp. 1–12 (2015). DOI 10.1007/s00371-015-1180-1
 26. Hasan, M., Samavati, F.F., Sousa, M.C.: Balanced multiresolution for symmetric/antisymmetric filters. *Graph. Models* **78**, 36–59 (2015). DOI <http://dx.doi.org/10.1016/j.gmod.2015.01.001>
 27. Hauser, H., Mroz, L., Italo Bischi, G., Gröller, M.: Two-level volume rendering. *IEEE Trans. Vis. Comput. Graph.* **7**(3), 242–252 (2001). DOI 10.1109/2945.942692
 28. Hsu, W.H., Ma, K.L., Correa, C.: A rendering framework for multiscale views of 3D models. *ACM Trans. Graph.* **30**(6), 131:1–131:10 (2011). DOI 10.1145/2070781.2024165
 29. Imanishi, K., Nakao, M., Kioka, M., Mori, M., Yoshida, M., Takahashi, T., Minato, K.: Interactive bone drilling using a 2D pointing device to support microendoscopic discectomy planning. *Int. J. Comput. Assist. Radiol. Surg.* **5**, 461–469 (2010)
 30. Kitware: Visualization toolkit (VTK) version 5.10.1. (2012). URL <http://www.vtk.org>
 31. Lee, T., Kashyap, R., Chu, C.: Building skeleton models via 3-D medial surface axis thinning algorithms. *CVGIP: Graph. Model. IM* **56**(6), 462–478 (1994). DOI 10.1006/cgip.1994.1042
 32. McGuffin, M.J., Tancau, L., Balakrishnan, R.: Using deformations for browsing volumetric data. In: Proceedings of the Conference on Visualization, VIS, pp. 401–408. IEEE Computer Society, Washington, DC, USA (2003). DOI <http://dx.doi.org/10.1109/VISUAL.2003.1250400>
 33. Pietriga, E., Appert, C.: Sigma lenses: Focus-context transitions combining space, time and translucence. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI, pp. 1343–1352. ACM, New York, USA (2008). DOI 10.1145/1357054.1357264
 34. Pindat, C., Pietriga, E., Chapuis, O., Puech, C.: JellyLens: content-aware adaptive lenses. In: Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology, UIST, pp. 261–270. ACM, New York, USA (2012). DOI 10.1145/2380116.2380150
 35. Renka, R.J.: Multivariate interpolation of large sets of scattered data. *ACM Trans. Math. Softw.* **14**(2), 139–148 (1988). DOI 10.1145/45054.45055
 36. Rezk-Salama, C., Scheuering, M., Soza, G., Greiner, G.: Fast volumetric deformation on general purpose hardware. In: Proceedings of the ACM SIGGRAPH/Eurographics Workshop on Graphics Hardware, HWWS, pp. 17–24. ACM, New York, USA (2001). DOI 10.1145/383507.383517
 37. Roberts, M., Packer, J., Sousa, M.C., Mitchell, J.R.: A work-efficient GPU algorithm for level set segmentation. In: Proceedings of the Conference on High Performance Graphics, HPG, pp. 123–132. Eurographics Association, Aire-la-Ville, Switzerland (2010)
 38. Ropinski, T., Viola, I., Biermann, M., Hauser, H., Hinrichs, K.: Multimodal visualization with interactive closeups. In: Proceeding of the Theory and Practice of Computer Graphics Conference, pp. 17–24. Eurographics Association (2009)
 39. Samavati, F.F., Bartels, R.H., Olsen, L.: Local B-spline multiresolution with examples in iris synthesis and volumetric rendering. In: S.N. Yanushkevich, M.L. Gavrilova, P.S.P. Wan, S.N. Srihari (eds.) *Image Pattern Recognition: Synthesis and Analysis in Biometrics, Series in Machine Perception and Artificial Intelligence*, vol. 67, pp. 65–102. World Scientific Publishing (2007)
 40. Sederberg, T.W., Parry, S.R.: Free-form deformation of solid geometric models. In: Proceedings of the 13th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH, pp. 151–160. ACM, New York, USA (1986). DOI 10.1145/15922.15903
 41. Shepard, D.: A two-dimensional interpolation function for irregularly-spaced data. In: Proceedings of the 23rd ACM National Conference, ACM, pp. 517–524. ACM, New York, USA (1968). DOI 10.1145/800186.810616
 42. Taerum, T., Sousa, M.C., Samavati, F., Chan, S., Mitchell, J.R.: Real-time super resolution contextual close-up of clinical volumetric data. In: Proceedings of the Eighth

- Joint Eurographics/IEEE VGTC Conference on Visualization, EUROVIS, pp. 347–354. Eurographics Association, Aire-la-Ville, Switzerland (2006). DOI 10.2312/VisSym/EuroVis06/347-354
43. Tiede, U., Schiemann, T., Höhne, K.H.: High quality rendering of attributed volume data. In: Proceedings of the Conference on Visualization, VIS, pp. 255–262. IEEE Computer Society, Los Alamitos, CA, USA (1998)
 44. Van Emmerik, M.J.: A direct manipulation technique for specifying 3D object transformations with a 2D input device. *Comput. Graph. Forum* **9**(4), 355–361 (1990). DOI 10.1111/j.1467-8659.1990.tb00427.x
 45. Wang, L., Zhao, Y., Mueller, K., Kaufman, A.: The magic volume lens: An interactive focus+context technique for volume rendering. In: Proceedings of the Conference on Visualization, VIS, pp. 367–374. IEEE Computer Society (2005). DOI 10.1109/VISUAL.2005.1532818
 46. Wang, S.W., Kaufman, A.E.: Volume sculpting. In: Proceedings of the 1995 Symposium on Interactive 3D Graphics, I3D, pp. 151–156, 214. ACM, New York, USA (1995). DOI 10.1145/199404.199430
 47. Westermann, R., Rezk-Salama, C.: Real-time volume deformations. *Comput. Graph. Forum* **20**(3), 443–451 (2001). DOI 10.1111/1467-8659.00537
 48. Winter, A.S., Chen, M.: Image-swept volumes. *Comput. Graph. Forum* **21**(3), 441–450 (2002). DOI 10.1111/1467-8659.t01-1-00604



Jeffrey F. Packer received his BSc and MSc degrees in Computer Science from the University of Calgary in 2010 and 2013, respectively. He is currently a Software Developer at GEO-SLOPE International Ltd. in Calgary, Canada, where he contributes to the architecture of finite element CAD software used for geotechnical engineering. Mr. Packer's research interests include non-photorealistic rendering (illustrative visualization and artistic rendering), real-time rendering, massively-data parallel computing, and games development.



Mahmudul Hasan is a PhD candidate in the Department of Computer Science of the University of Calgary, where he received his MSc degree in Computer Science in 2009. He graduated with a BSc degree in Computer Science from the Independent University, Bangladesh (IUB), as a *Summa Cum Laude*, ranking 1st among the graduating class of 170 students from his school in 2005. Mr. Hasan was the *Posters Chair* of the 39th

Graphics Interface (GI) conference in 2013 and the *Organizing Co-chair* of the 3rd Workshop on Digital Earth held in Banff, Canada in 2015. His research interests include multiresolution and visualization in computer graphics.



Faramarz F. Samavati is a Professor of Computer Science at the University of Calgary. His research interests include computer graphics, geometric modeling, visualization, and 3D imaging. He has published more than 100 papers, one book, and holds two patents. He is an Associate Editor of Elsevier's *Computer & Graphics* journal and was a Principal Investigator and a Project Leader of the Network of Centers of Excellence (NCE) of Canada in Graphics, Animation and New Media (GRAND). In the years of 2011 to 2015, he has received five Best Paper Awards, a Digital Alberta Award, a Great Supervisor Award, and a University of Calgary Award that honors his contribution in the development of new technologies and innovations.